TDB-ACC-NO: NN8902351

DISCLOSURE TITLE: Programmable Memory Controller

PUBLICATION-DATA: IBM Technical Disclosure Bulletin, February 1989, US

VOLUME NUMBER: 31

ISSUE NUMBER: 9

PAGE NUMBER: 351 - 354

PUBLICATION-DATE: February 1, 1989 (19890201)

CROSS REFERENCE: 0018-8689-31-9-351

DISCLOSURE TEXT:

- A technique is described whereby a programmable memory controller (PMC) provides a design support means to rapidly implement high speed dynamic memory devices into computer designs. The PMC provides the design support by enabling a designer to alter processor-memory timing relationships through the use of microcode so that design changes and upgrades to memory circuitry can rapidly be made by simply altering a memory initialization table. During computer product development cycles, faster and denser dynamic random-access memory (RAM) devices become available which have a tendency to obsolete predecessor RAM chips. A designer, using the faster chips, must make extensive design changes, often requiring a new product design cycle, to implement the advanced memory in the memory circuitry.

The concept described herein provides a method whereby the designer need only re-program the circuitry involved in order to be able to accept a new memory technology. The only hardware changes would be a new read-only memory (ROM) for configuring and testing the additional memory and a new circuit layout to accept the new RAM modules because of different pin configurations. The PMC consists of three basic components:

1) A Critical Timing Control Signal Generator - Allows the circuit designer to program processor-memory timings for each memory **interface** signal. ***** SEE ORIGINAL DOCUMENT *****

2) A Programmable Address Multiplexer - Allows the circuit designer to program the width of the RAM multiplexed address bus.

- 3) A Programmable/Variable Refresh Timer - Allows the designer to compensate for variations in refresh rates of the various memory devices. ***** SEE ORIGINAL

DOCUMENT ***** The critical timing control signal generator is composed of **delay** line 10, as shown in Fig. 1, **delay** line selector 11 and **register** 12, which is used to retain the **programmed delay** value. The generator enables the designer to define the time between critical timing points, such as the **delay** between memory timings. A critical timing point may require more than one critical timing control signal generator. The programmable address multiplexer consist of a series of 4 to 1 data selects for address control, 2 to 1 data selectors for RAM control, refresh control logic unit 13, as shown in Fig. 2, and refresh timer address counter 14.

The physical width of this bus is A0 - A10, so as to allow memory modules that can address up to 4M **words**. This allows the designer to migrate their circuit designs to denser memory chips, as they become available. Refresh control logic unit 13 and refresh timer address counter 14 provide the mechanism for refreshing memory modules at different refreshing rates. The timing chart of Fig. 3 illustrates the transitions produced by each memory signal. For example, the RAS signal requires a falling edge; therefore, the microcode is required to write a "2" into the **delay** line selector that controls the falling edge of RAS. For the rising edge of the RAS, the microcode should write a "13" into the **delay** line selector that controls the rising edge of RAS.

Within the data area of the ROM, a table would be generated, such as the following:

```
DC      A(Delay_Line_Selector_RAS)
DC      X'0213'
DC      A(Delay_Line_Selector_CAS1)
DC      X'0700'
DC      A(Delay_Line_Selector_CAS2)
DC      X'2314'
DC      A(Delay_Line_Selector_MOE)
DC      X'0800'
DC      A(Delay_Line_Selector_MW)
DC      X'2415'
DC      A(Delay_Line_Selector_DTACK)
DC      X'0713'
DC      A(Delay_Line_Selector_ADDRMUX)
DC      X'0514' A simple microcode loop can be written to
```

take the address off the table and use it to write the following two bytes as a **word** into the two **delay** line selectors.

The falling edge **delay** line selector would use the upper byte of data, while the lower byte of data would be written into the rising edge **delay** line selector. By using the PMC, the designer will have the ability to alter timing relationships, through the use of microcode. As new technology and design changes occur, a simple alteration of the memory initialization table within the ROM will enable rapid implementation of the new technology, or a design change, to the

computer design.

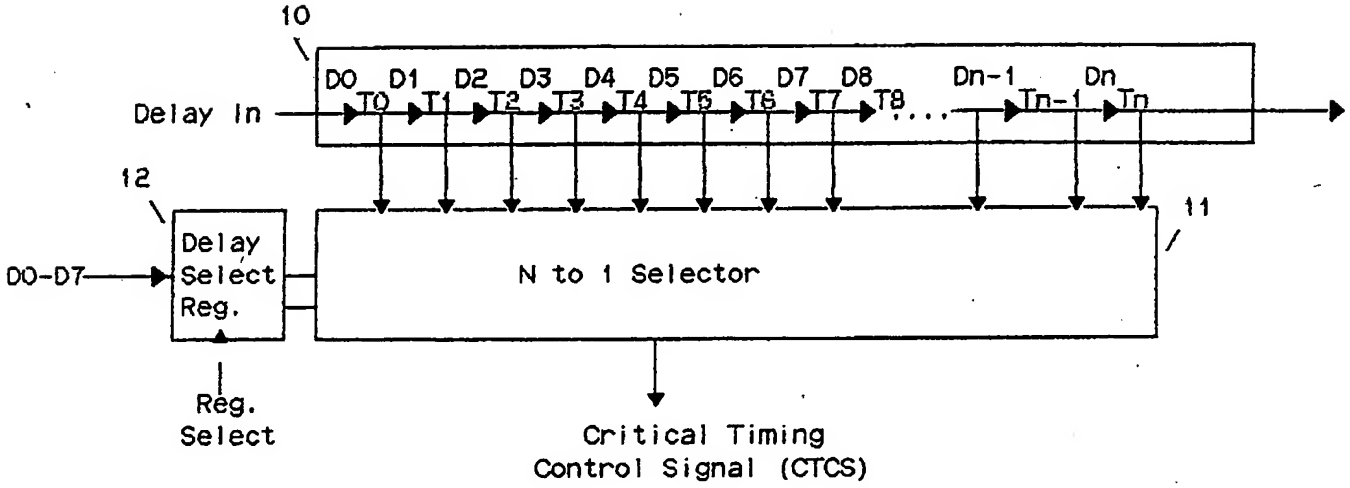
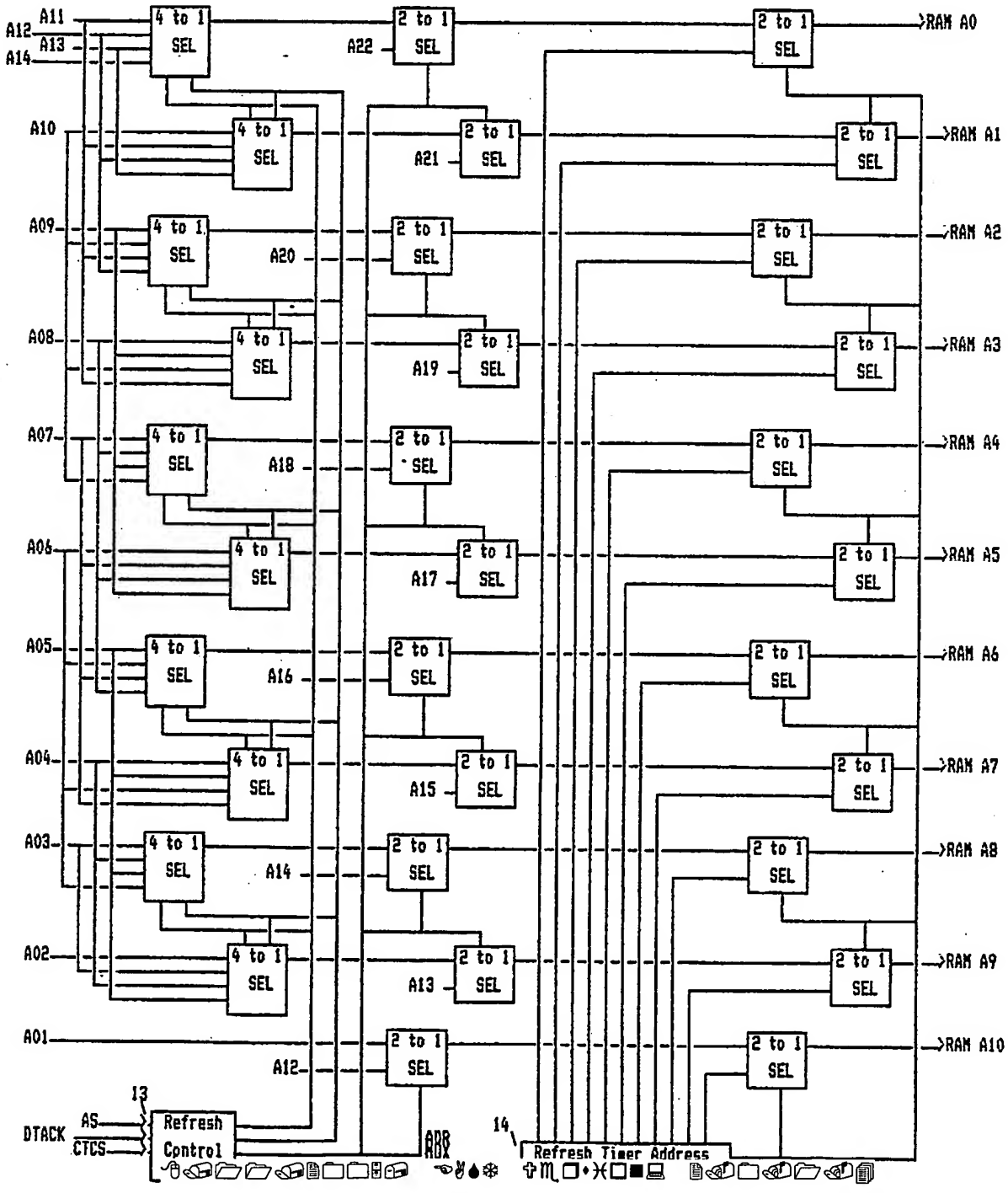

10
D0 D1 D2 D3 D4 D5 D6 D7 D8 Dn-1 Dn
Delay In
T0 T1 T2 T3 T4 T5 T6 T7 T8 ... Tn-1 Tn
12
11
Delay Select Reg.
D0-D7
N to 1 Selector
Reg. Select
Critical Timing Control Signal (CTCS)

Fig. 1


A11
A12
A13
A14
A10
A09
A08
A07
A06
A05
A04
A03
A02
A01
4 to 1 SEL
A22
A21
A20
A19
A18
A17
A16
A15
A14
A13
A12
2 to 1 SEL
RAM A0
RAM A1
RAM A2
RAM A3
RAM A4
RAM A5
RAM A6
RAM A7
RAM A8
RAM A9
RAM A10
13
DTACK
AS
CTCS
Refresh Control
ARB ADX
14
Refresh Timer Address

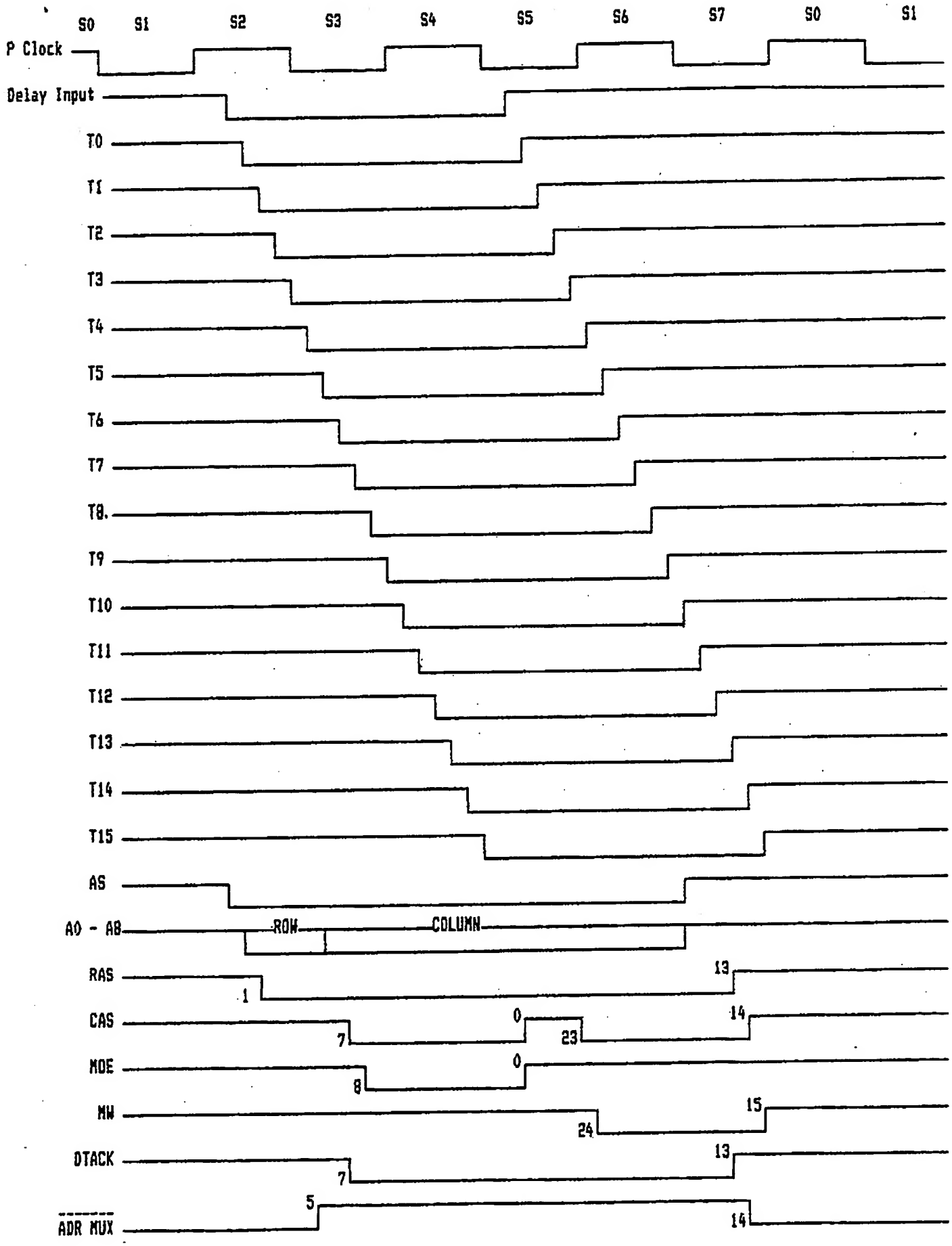

S0
S1
S2
S3
S4
S5
S6
S7
S0
S1
P Clock
Delay Input
T0
T1
T2
T3
T4
T5
T6
T7
T8.
T9
T10
T11
T12
T13
T14
T15
AS
A0 - A8
ROW
COLUMN
RAS
1
13
CAS
7
0
23
14
MOE
8
0
MW
24
15
DTACK
7
13
ADR MUX
5
14

Fig. 3